Annals of the University of Craiova, Electrical Engineering series, No. 33, 2009; ISSN 1842-4805

7TH INTERNATIONAL CONFERENCE ON ELECTROMECHANICAL AND POWER SYSTEMS
October 8-9, 2009 - Iaşi, Romania

# EXPERT SYSTEM FOR RELIABILITY ANALYSIS OF ELECTRICAL SYSTEMS USING THE TREE OF FAULTS METHOD

**Virginia IVANOV*, Maria BROJBOIU*, Sergiu IVANOV****

**Faculty for Electrical Engineering,* ***Faculty for Electromechanical Engineering,*
*University of Craiova, vivanov@elth.ucv.ro*

***Abstract*** **– The expert systems are products of computer science research called Artificial Intelligence which follows the development of intelligent programs. These kinds of techniques are successfully promoted in industry. In the future, the treatment of the engineering problems with specific methods of the artificial intelligence shall considerable increase. The paper present an application of the expert systems is the case of reliability analysis of electrical systems by using the tree of faults method.**

***Keywords***: *expert system, reliability, tree of faults.*

## 1. INTRODUCTION

An expert system is a program which follows the knowledge, rations for obtaining the results in a difficult activity, usually done only by the human experts. If a human expert has knowledge in a specific area, an expert system uses the knowledge stored in a knowledge base. This base unifies the knowledge associated to a specific domain. The human experts ration and get the conclusions based on the knowledge they have. The expert systems ration by using the knowledge stored in the knowledge base.
The systems based on knowledge can be applied for any area of knowledge. Expert systems must contain three main modules [1] (Fig. 1):

a) *The knowledge base* is done from the sum of specific knowledge specified by the human expert. The knowledge loaded here is mainly the description of the objects and of the relations between them.
b) *The inference devices* consist in the sum of the algorithms for determining solutions for the expertise problems, similarly to the human expert.
c) *The base of facts* (Factual knowledge) contains a dynamic collection of information which changes itself during the call of the expert system. It depends on the practical expertise problem.
Besides these modules, an expert system contains also several modules which offer the ability of communication with the user and the human expert.
The user interface is the one which performs the dialogue between the user and the system, by using a quasi-natural language. It generally contains the systems of menus and the graphical user interfaces specific to the men-machine communication.
*Knowledge acquisition module* performs the task of acquiring the specialized knowledge offered by the human expert or by the knowledge engineer. It verifies the validity of the knowledge and generates a knowledge base specific to the expert system.
*Explanations module* allows tracing the way followed during the ration activity by the expert system. It outputs arguments for the resulted solutions.

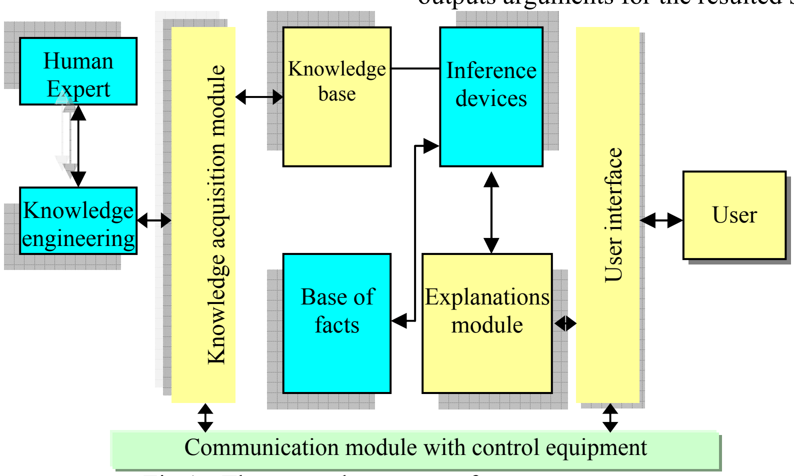



Fig.1 - The general structure of an expert system

## 2. EXPERT SYSTEM FROM RELIABILITY ANALYSIS OF ELECTRICAL SYSTEMS

One application of the expert systems is the case of reliability analysis of electrical systems by using the tree of faults method [2].

The analysis of the systems reliability by using the tree of faults method consists in several steps: system definition; making up of the tree of faults with regard to the considered critical event; quantitative and qualitative evaluation of the tree of faults.

In order to characterize the system, the functional diagram of the system is used. The subsystems and elements are identified, as well as the functional connections between them.

The realization the tree of faults is an ample process in which the analyst must prove good and deep knowledge of the studied system. The tree of faults is developed hierarchically starting from the top level which is the considered critical event. Then the secondary level is developed, in accordance with the desired level of detail.

The quantitative analysis of the tree of faults means that, starting from the probability of the events occurrence and taking into account their propagation method, by the way of the tree logic gates, the probability of occurrence of the considered fault is determined.

The method of the tree of faults has as main disadvantage the difficulty to build the tree of faults for highly complex systems. For this kind of systems, the methodology based on the tree of faults may be quite difficult to be performed manually.

For applying this method, a electrical system (Fig.2) is considered. The analyzed failure is the overheating of the servomotor.

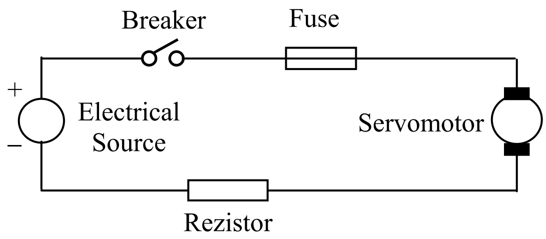



Fig.2 - Example of an electrical system

The boundary conditions of the analyzed system are:
- Initial condition – breaker closed;
- Not permitted events – faults due to external factors.

### 2.1 The primary fault tree

Starting from the considered critical event, based on the deductive reasoning which implies that the servomotor will overheat due to an electrical overload or due to a primary fault of the servomotor (increasing of the friction in bearings, or a winding fault), the primary fault tree of the analyzed system can be developed (Fig. 3).

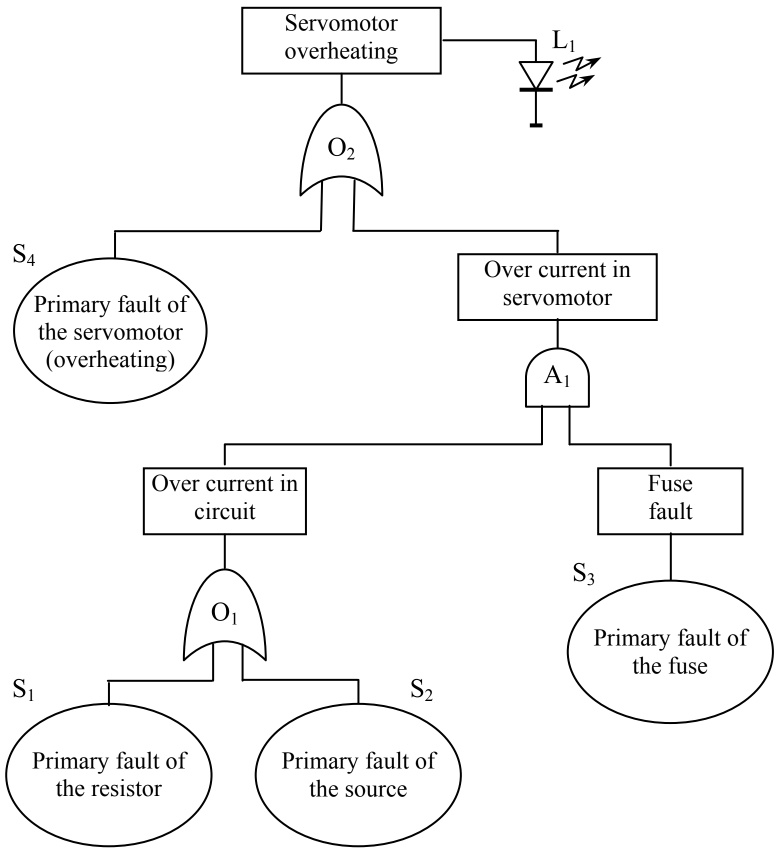



Fig. 3 – Primary fault tree of the analyzed

The event “Over current in servomotor” occurs if it is over current in the circuit and the fuse does not operate.
The event “Over current in circuit” occurs if a fault like short circuit of the resistor, or fault of the source (over voltage) occurs.
Thus, the fault tree is completely defined for the primary faults.
The implementation of the fault tree is performed with logic gates. Each possible fault which influences the considered critical event is represented by a source. The logic 1 of each source means existence of the corresponding fault.
In order to reduce the true table of the tree corresponding to the considered fault, an expert system was developed, by using the programming language CLIPS [3].
The specific algorithm developed in CLIPS language simplifies the table of truth for a logical complex circuit with more inputs (sources) and outputs.
The simplification procedure implies the following steps:
- the connections between the circuit components are initialized;
- the response of the system when all sources are set to zero is determined;
- a single source is modified and the answer of the system is determined. By using the Gray code, all possible combinations of inputs are iterated. By using the Gray code, only one source is modified at each step, in order to determine the answer in the table of decisions (the use Gray code determines the minimization of the execution time);
- during the determination of the responses, a rule checks if two sets of inputs which are different by a single input determines the same answer. If YES, this single input can be replaced by „*” (it signifies that the value of that input has no importance for obtaining the same answer);
- once that all the answers and the simplifications were determined, the table of decisions of the circuit is printed.
This application exemplifies the use of most usual procedures available within CLIPS software how interesting can be integrated with the rules.
The sources which influence the considered critical event are:
$S_1$ – Primary fault of the resistor;
$S_2$ – Primary fault of the source;
$S_3$ – Primary fault of the fuse;
$S_4$ – Primary fault of the servomotor.
The results of running the program (Fig. 4) highlight the simplification of the true table, the 16 (24) possible combinations of the inputs being reduced to 5 distinct combinations.

```
CLIPS> Defining definstances: circuit
Redefining deffunction: connect-circuit
TRUE
CLIPS> (reset)
CLIPS> (run)
 S-1  S-2  S-3  S-4  |  L-1
---------------------+------
  0    0    1    0   |  0
  *    *    0    0   |  0
  *    1    1    0   |  1
  1    0    1    0   |  1
  *    *    *    1   |  1
CLIPS> |
```

Fig. 4 – The results of the program for the primary tree

### 2.2 The secondary fault tree

For marking out how the secondary fault tree are developed (the secondary faults are due to the interactions between the system components), the same system will be analyzed, but for another critical event, the no operation of the servomotor. The other boundary conditions are the same as previous.
Starting from the considered critical event, the secondary fault tree of the analyzed system can be belt (Fig. 5).
For this case, the sources which influence the considered critical event are:
$S_1$ – Primary fault of the resistor (short circuit);
$S_2$ – Primary fault of the source (over volatge);
$S_3$ – Primary fault of the fuse;
$S_4$ – Primary fault of the servomotor;
$S_5$ – Primary fault of the source;
$S_6$ – Primary fault of the resistor (open);
$S_7$ – Primary fault of the breaker;
$S_8$ – Breaker open.
Due to the fact that the deductive reasoning is evident by analysing the structure of the fault tree, following will be set off only the particularities specific to some symbols. Thus, the rhombus symbol was used for specifying that the event – open breaker – is not developed until the causes are known. The open breaker is an external fault regarding the considered limits of the system. For this analysis there is not enough information for its development.
Taking into account the great number of sources (inputs), the total number of possible combinations being 256 ($2^8$), the expert system performs algorithms difficult to be done manually. The results of running the program (Fig. 6) highlight the simplification of the true table, the 256 possible combinations of the inputs being reduced to 9 distinct combinations.

## 3. QUANTITATIVE ANALYSIS OF THE RELIABILITY OF THE ELECTRICAL SYSTEMS

The quantitative analysis of the fault tree is the last step in the methodology of reliability analysis of the systems by using the fault tree method.

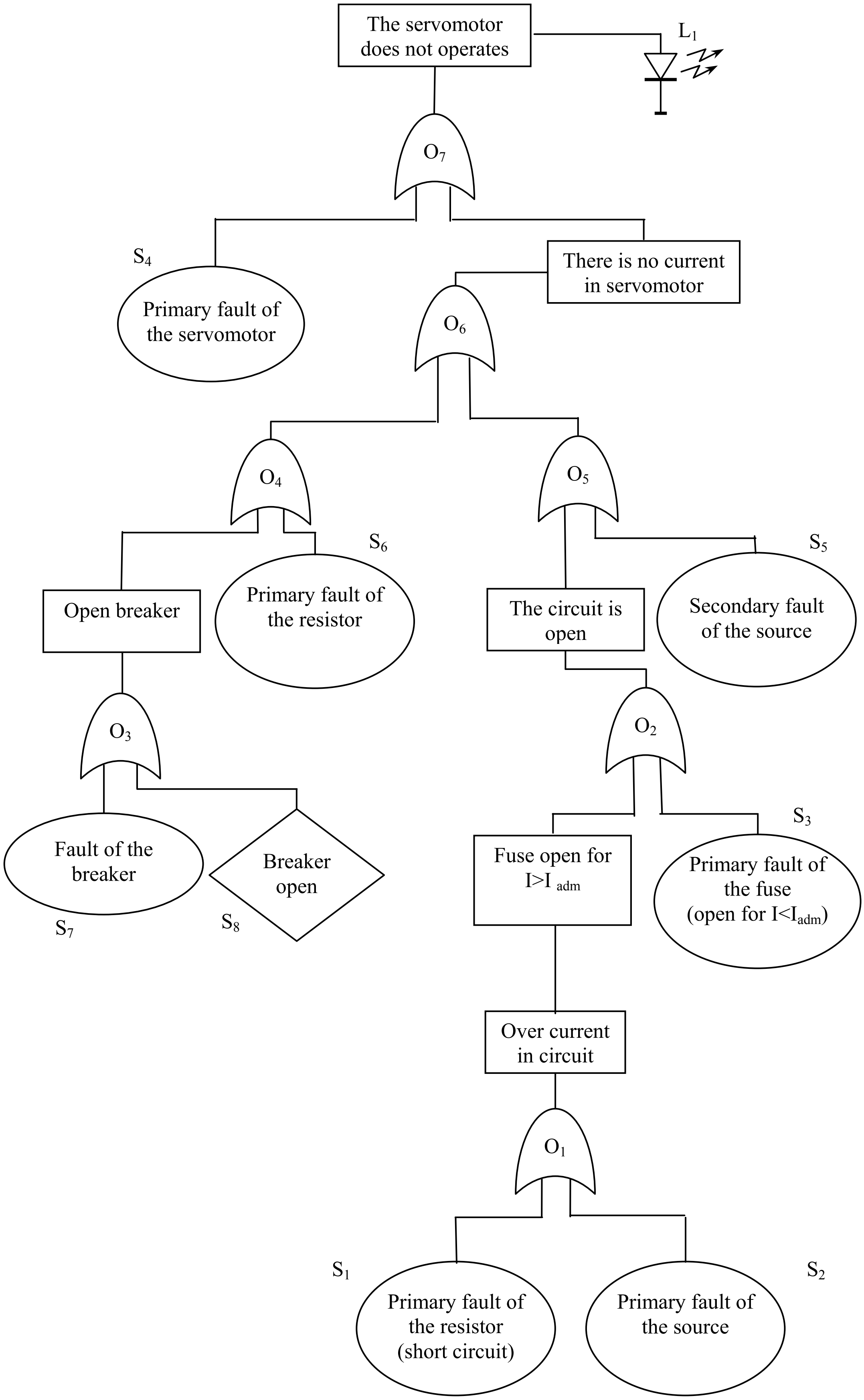
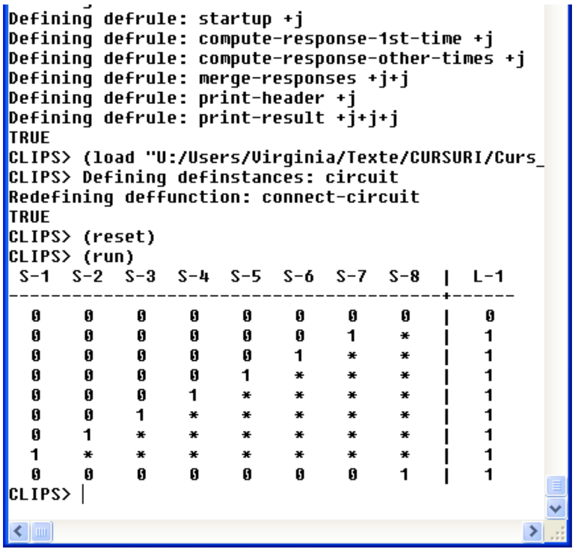



Fig.5 – Secondary fault tree



```
Defining defrule: startup +j
Defining defrule: compute-response-1st-time +j
Defining defrule: compute-response-other-times +j
Defining defrule: merge-responses +j+j
Defining defrule: print-header +j
Defining defrule: print-result +j+j+j
TRUE
CLIPS> (load "U:/Users/Virginia/Texte/CURSURI/Curs_
CLIPS> Defining definstances: circuit
Redefining deffunction: connect-circuit
TRUE
CLIPS> (reset)
CLIPS> (run)
 S-1  S-2  S-3  S-4  S-5  S-6  S-7  S-8  |  L-1
-----------------------------------------+------
  0    0    0    0    0    0    0    0   |  0
  0    0    0    0    0    0    1    *   |  1
  0    0    0    0    0    1    *    *   |  1
  0    0    0    0    1    *    *    *   |  1
  0    0    0    1    *    *    *    *   |  1
  0    0    1    *    *    *    *    *   |  1
  0    1    *    *    *    *    *    *   |  1
  1    *    *    *    *    *    *    *   |  1
  0    0    0    0    0    0    0    1   |  1
CLIPS> |
```

Fig.6 – The results of the program for the secondary tree

This impose to determine the probability of occurrence of the critical event analyzed, starting from the probabilities of occurrence of the events (faults) and taking into account their propagation ways through the logic gates of the tree.

This methodology imposes to write the structural logic function starting from the fault tree of the analyzed system as::

$$E_{cr} = \Phi_l\left(E_1, E_2, ..., E_i, ...E_n\right) \ , \qquad (1)$$

where,

$E_{cr}$ is the critical event of the system, expressed in terms of the primary events $E_i$ (i=1, 2,…,n), considered as independent between them.

Starting from the logic function (1) it can be expressed the algebraic one

$$E_{cr} = \Phi_a\left(E_1, E_2, ..., E_i, ...E_n\right) \ . \qquad (2)$$

By taking into account the transformations specified in Table 1, corresponding to the basic logic gates of the fault trees:

Table 1

| Logic gate | Relation | |
|---|---|---|
| | logical | Algebraic |
| AND | $E_i \cap E_j$ | $E_i \cdot E_j$ |
| OR | $E_i \cup E_j$ | $E_i + E_j - E_i \cdot E_j$ |

Even the method described above is systematic, it is quite complex because it requires writing and processing the structural function corresponding to the fault tree of the analyzed system. In practice, in order to facilitate the quantitative evaluation, it is possible to avoid the writing of the structural function. In this case, the calculus will be done step by step, from down to up, starting from the basic levels corresponding to the primary events, to the critical event. Following will be presented the relations which allow to highlight the propagation of the tree events by the way of the fundamental logic gates (AND, OR).

For an AND gate with $n$ inputs we have:

$$P\left(1 \cap 2 \cap ... \cap n\right) = P\left(1\right) \cdot P\left(2\right) \cdot ... \cdot P\left(n\right) \ . \qquad (3)$$

For an OR gate with two inputs we have:

$$P\left(1 \cup 2\right) = P\left(1\right) + P\left(2\right) - P\left(1\right) \cdot P\left(2\right) \ . \qquad (4)$$

For small enough fault probabilities (in practice $P < 10^{-2}$, which is quite usual), it can be used the approximate expression:

$$P\left(1 \cup 2\right) = P\left(1\right) + P\left(2\right) \ . \qquad (5)$$

The simplified expression (5) can be generalized for an OR gate with $n$ inputs:

$$P\left(1 \cup 2 \cup ... \cup n\right) = P\left(1\right) + P\left(2\right) + ... + P\left(n\right) \ . \ (6)$$

### 3.1 Primary fault tree

The primary fault tree from Fig.3 will be quantitatively evaluated. The probability of occurrence of the event $E_{cr}$ will be estimated, starting from the probabilities of occurrence of the primary events $E_1$, $E_2$, $E_3$, $E_4$. A possible approach is to write the structural logic function corresponding to the tree [4]:

$$E_{cr} = \left[\left(E_1 \cup E_2\right) \cap E_3\right] \cup E_4 \ . \qquad (7)$$

For the presented example, thanks to the reduced complexity of the structural function, the probability of occurrence of the critical event can be expressed directly:

$$P\left(E_{cr}\right) = P\left\{\left[\left(E_1 \cup E_2\right) \cap E_3\right] \cup E_4\right\} \ . \qquad (8)$$

It results the algebraic expression corresponding to relation (8):

$$P\left(E_{cr}\right) = \left\{\left[P\left(E_1\right) + P\left(E_2\right)\right] \cdot P\left(E_3\right)\right\} + P\left(E_4\right) \ .(9)$$

### 3.2 Secondary fault tree

In order to quantitatively analyze the secondary tree from Fig. 5, it is better to avoid writing the structural logic function, because it would be too difficult to process it. The probability of occurrence of the critical event can be evaluated by computing step by step, considering the expressions (3), (5), (6). This approach is exemplified in Fig. 7, where is computed the probability of occurrence of the critical event, staring from the probabilities of occurrence of the primary events. The initial data for the probabilities of the primary events can be obtained from the datasheets of the components.

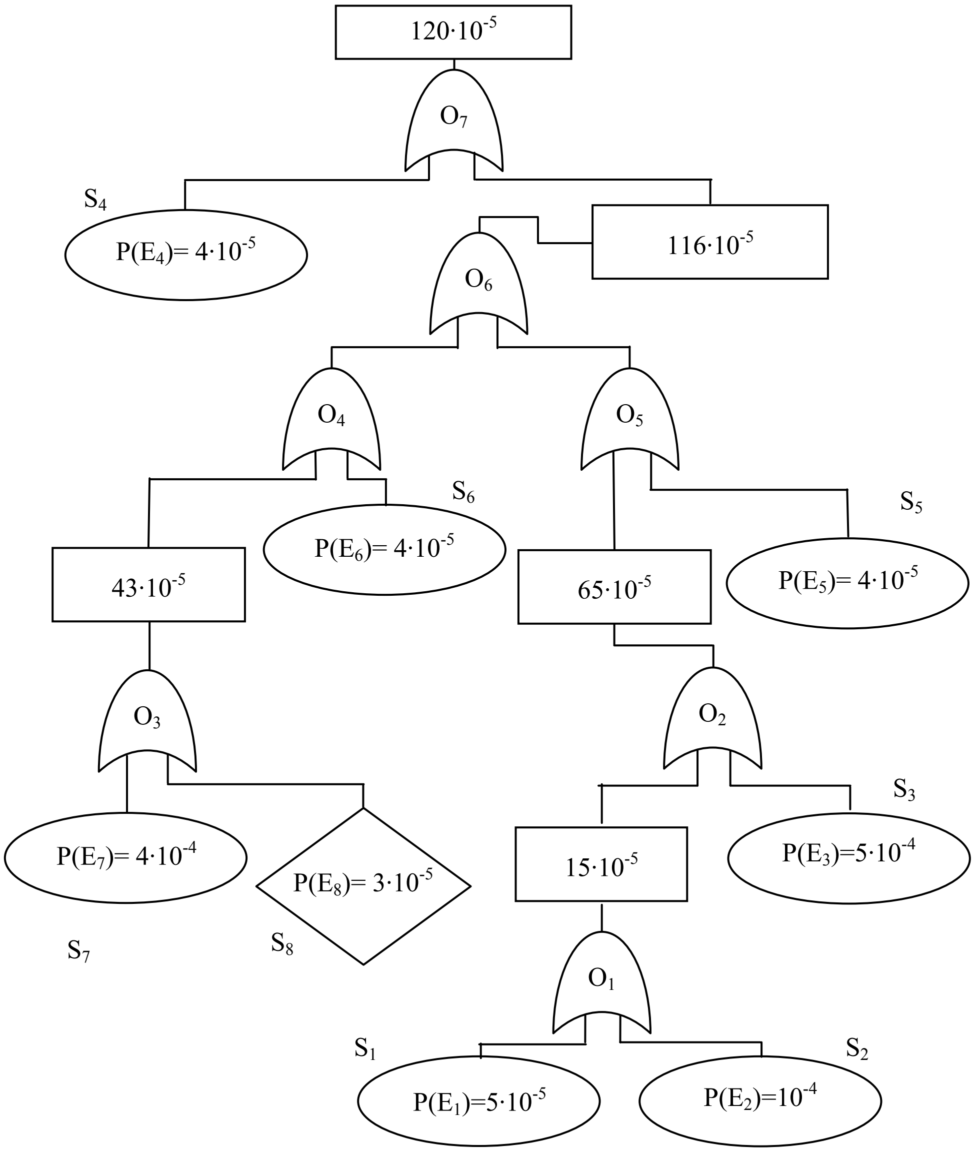



Fig. 7 - Computing the fault probability of the secondary tree

## 4. CONCLUSIONS

This paper presents an application of the expert systems for the quantitative and qualitative analysis of the reliability of the electric systems. In order to achieve the results, the fault tree method is used. Taking into account the great number of possible faults (*n* sources), the total number of possible combinations being $2^n$, the expert system performs algorithms difficult to be done manually. This example shows how the most usual procedures available in CLIPS can be used and interesting they can be integrated with the rules.

## References

[1] D.W. P a t t e r s o n, *Introduction to Artificial Intelligence and Expert Systems*, Prentice-Hall International, 1999.

[2] C.Y. T e o, *A comprehensive fault diagnostic System using artificial intelligence for sub-transmission and urban distribution networks*, IEEE Transactions on Power Delivery, Vol. 12, Nr. 4, November 1997.

[3] CLIPS - Reference Manual, Volume I, *Basic Programming Guide*, Version 6.10, 1998.

[4] V. Ivanov, M. Brojboiu, S. Ivanov, *The reliability analysis of electrical grids using expert systems*, Buletinul Institutului Politehnic Iaşi, Tomul LIV(LVIII), Fasc. 3 Electrotehnică, Energetică, Electronică, ISSN 1223-8139, pp.1-4, 2008.

[5] V. Ivanov, *Sisteme de supraveghere a echipamentelor electrice*, Editura Universitaria, Craiova, 2007.